Towards Deriving Theories from Data: Frontiers for Model Inference in Astro-&Geophysics

NASA AI Workshop, November 26, 2018

Victor Pankratius

Massachusetts Institute of Technology
Kavli Institute for Astrophysics and Space Research

Email: pankrat@mit.edu
Web: victorpankratius.com

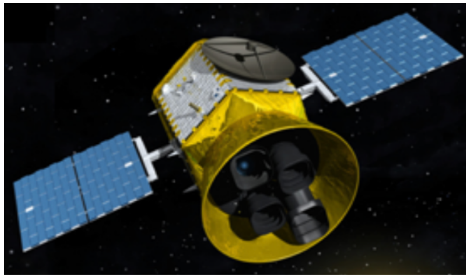
AIST16 80NSSC17K0125
PI Pankratius

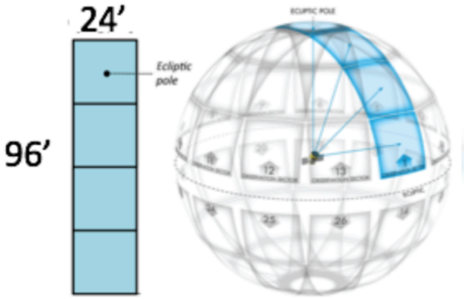
ACI1442997
PI Pankratius

Overview

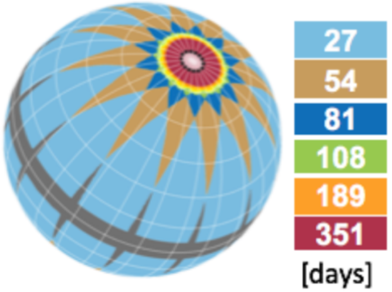
- Discuss AI in science – now and in the future
- Based on two examples:
  - Astrophysics: Exoplanet search
  - Geophysics: Earth deformation, volcanoes

Exoplanet Search





4x 2048x2048 CCDs



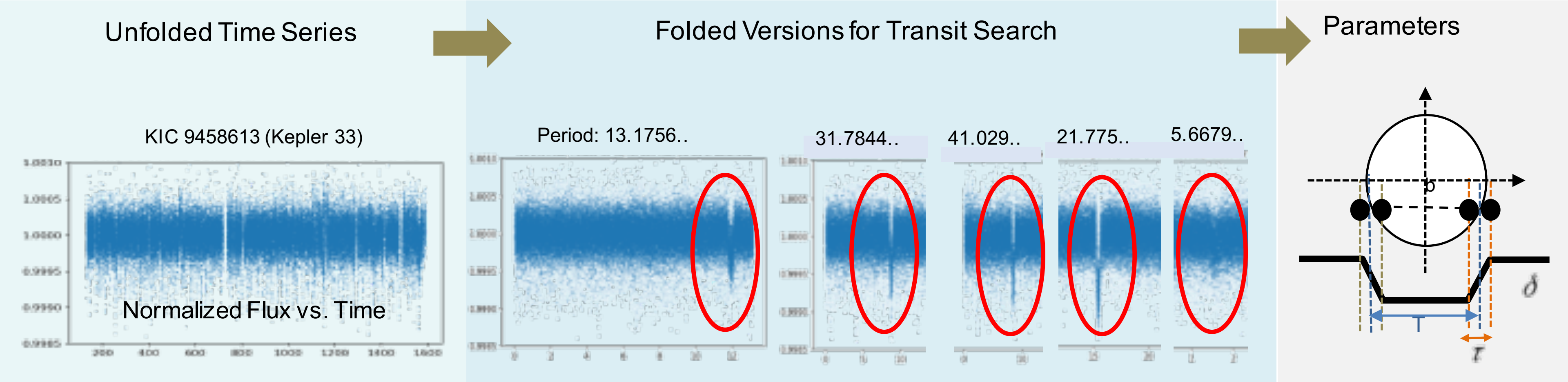
Transiting Exoplanet Survey Satellite (TESS)

- Near all-sky survey
- Launched April 18, 2018
- Kepler mission follow-up, stars 10-100 brighter
- Expecting thousands of new exoplanets smaller than Neptune and potentially dozens that are comparable to our Earth
- Full frame images every 30 minutes, 200,000 pre-selected stars monitored with 2 min cadence
- TESS processing pipeline extracts light curves
- Problems similar to future Big Data applications, e.g., Large Synoptic Survey Telescope (LSST) and others

[https://tess.mit.edu; https://tess.gsfc.nasa.gov; Ricker14]

Exoplanet Search

Transit Search: State-of-the-art



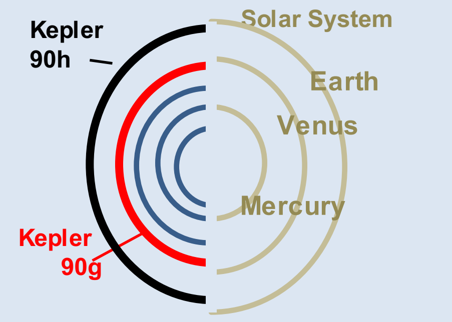
→ Machine learning and other methods typically applied on folded light curves [Shallue18]

[Kovacs02, Seager11, Winn14]

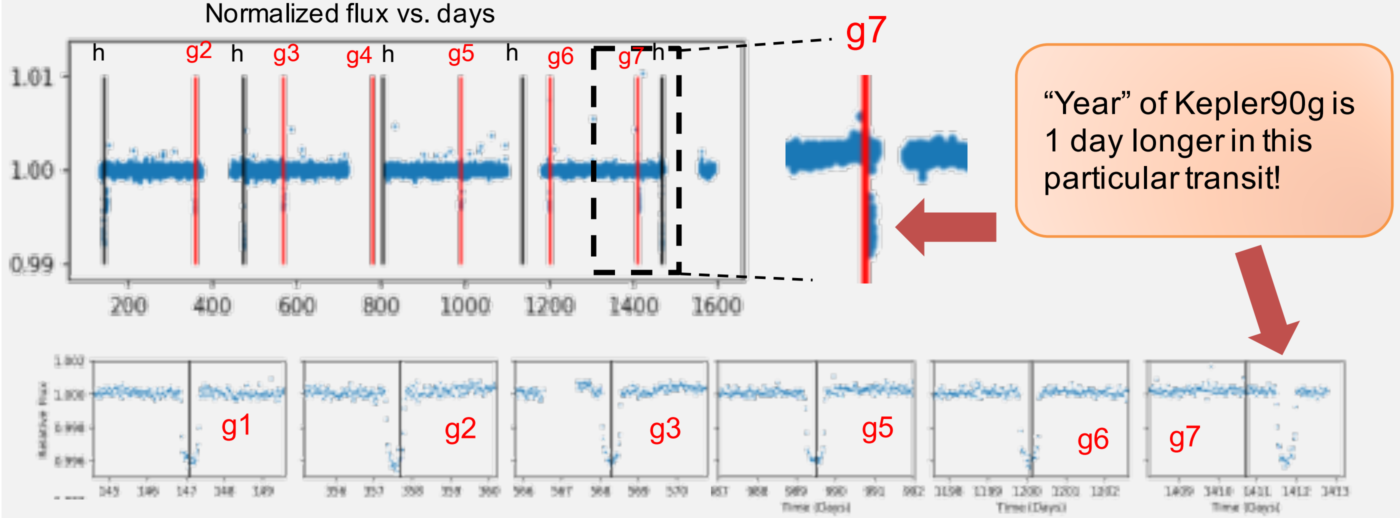
Exoplanet Search

However, there is more information in the unfolded time series.

→ Revealing irregular Transit Timing Variations (TTV) in Kepler90 system



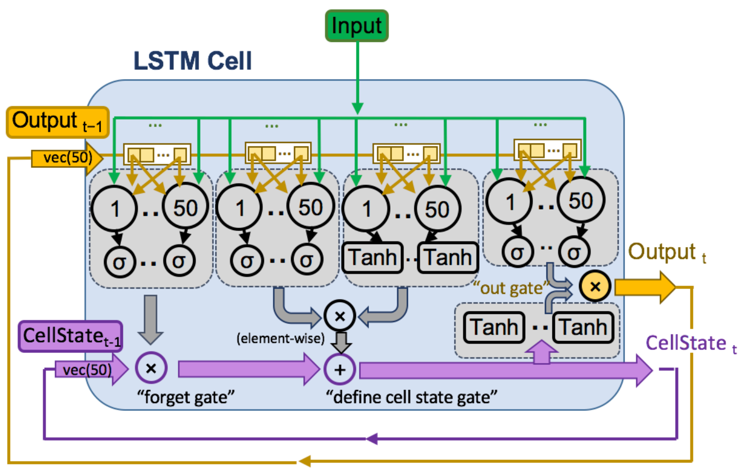
Period [days]	h: 331.6	g: 210.6
Mass [Jupiter Masses]	h: <1.2	g: <0.8

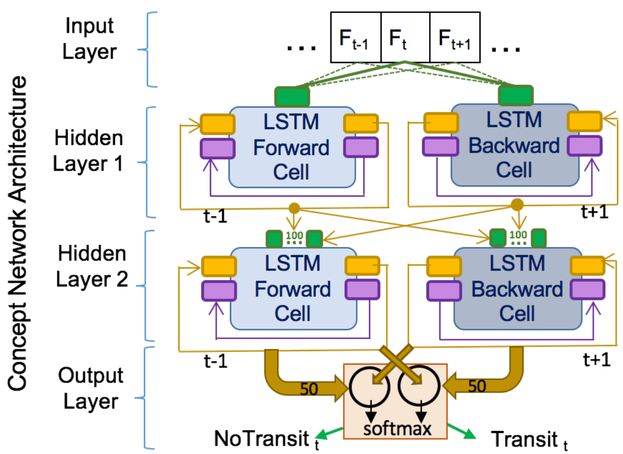


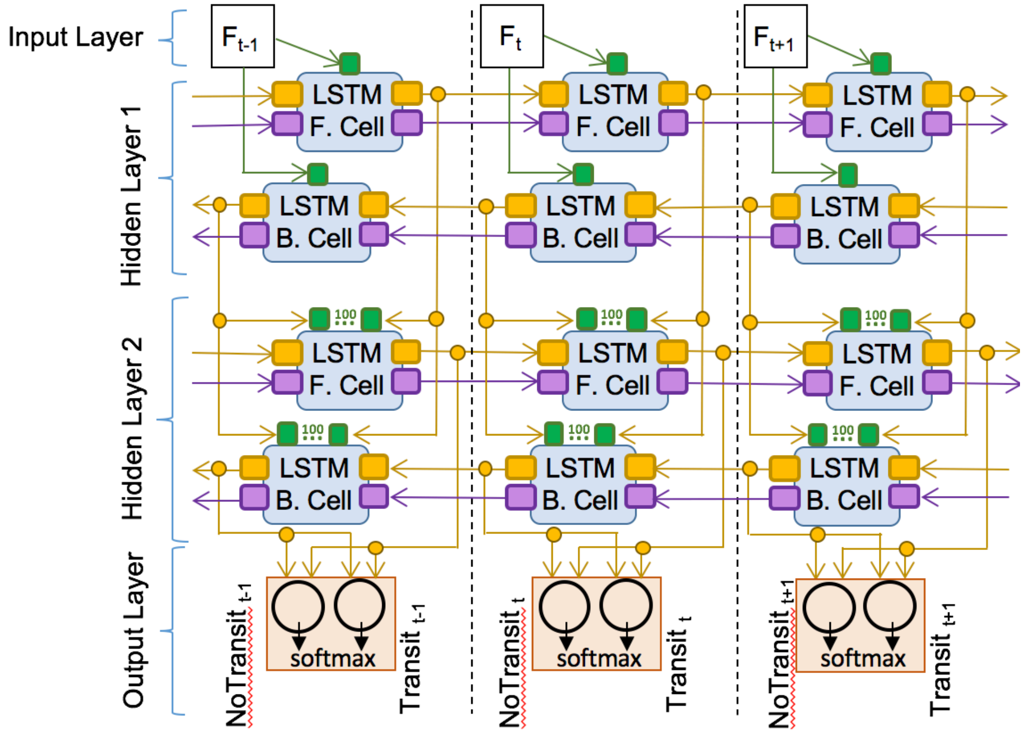
“Zooming” in on transits; red & black lines = catalog-listed periods

Bi-directional LSTM Networks in Exoplanet Search

A Toy Example:







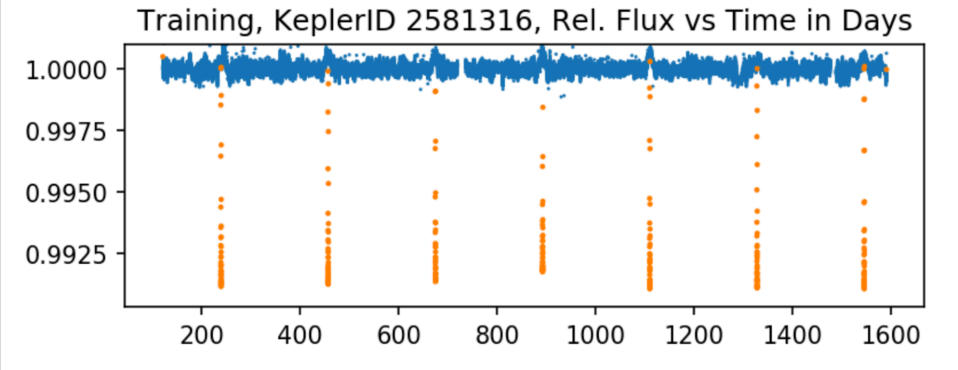
Networks that are "deep" in time

Victor Pankratius

Massachusetts Institute of Technology

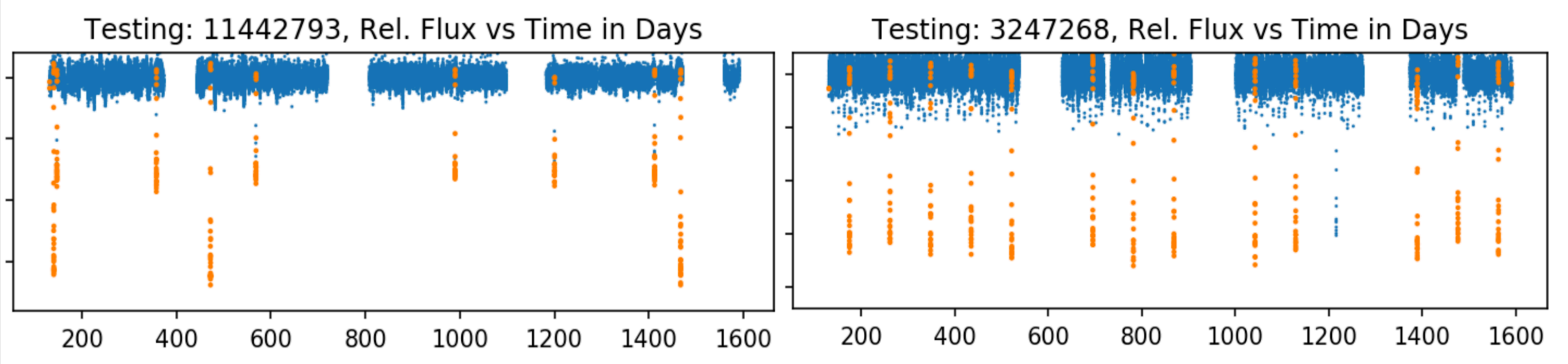
Bi-directional LSTM Networks in Exoplanet Search

BDLSTM example: learning **planet transits**



Applying trained BDLSTM to other light curves

[training: 50 epochs, 1 second steps, 0.5 dropout rate, until accuracy = 0.9797]



Massachusetts Institute of Technology

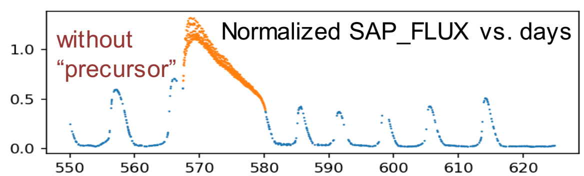
Bi-directional LSTM Networks: Other Phenomena

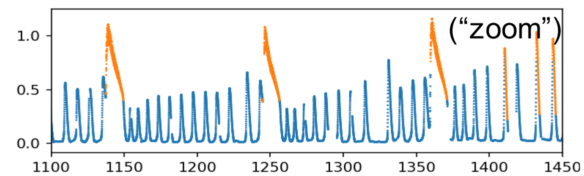
Variable Star Phenomena: **Learning Dwarf Nova Events**

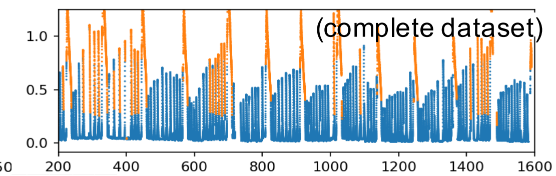
Example: V344 Lyr (Kepler 7659570)

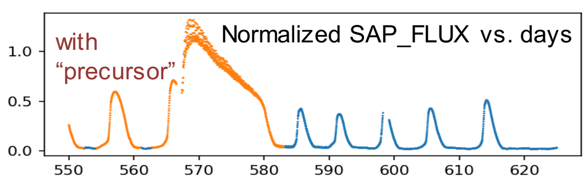
Training set = 1 piece of time series

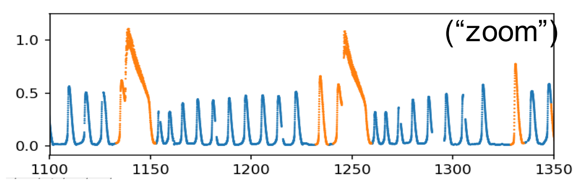
Preliminary BDLSTM Prediction on Test Set (rest of time series)

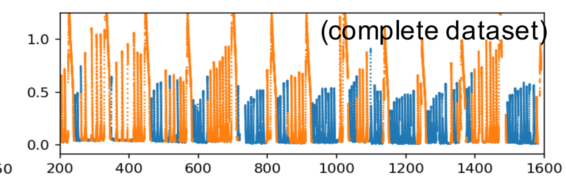












Note: potentially useful prediction capability based on empirically learned model

Next: Establishing Data – Model Connections

What do humans typically do?

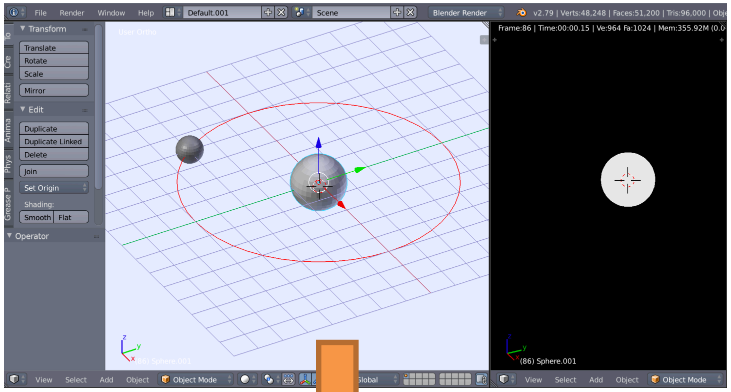
- Look at light curve → develop a "mental model" (hypothesized planetary system, related phenomenon)
- "Play" in imagination, unfold over time
- Anticipate dynamics
- Look back at the light curve for supportive clues

→Inverse problem solved iteratively by generating multiple forward models + pruning those that do not exhibit the right properties

→This process can be automated

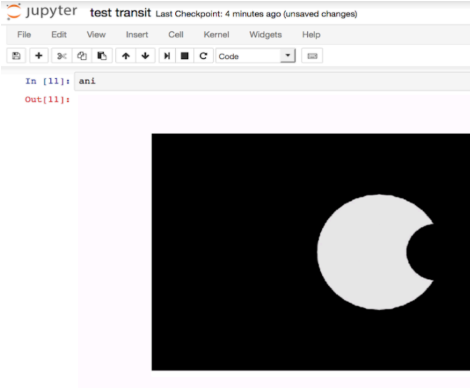
Next: Establishing Data – Model Connections

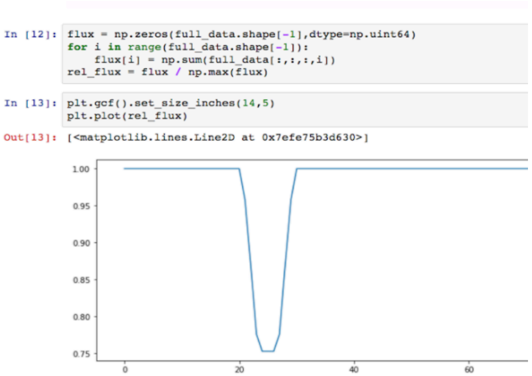
Proof of concept example:

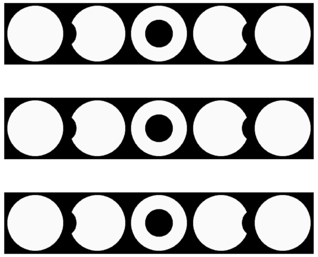


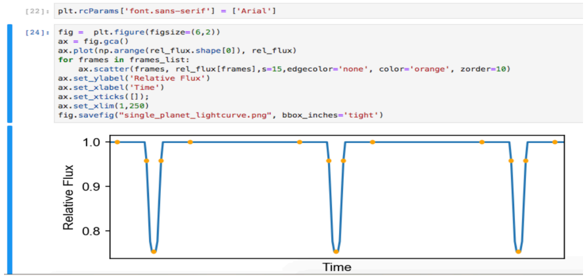
blender.org Raytracer

Programmatic Interface in Python Jupyter Notebooks







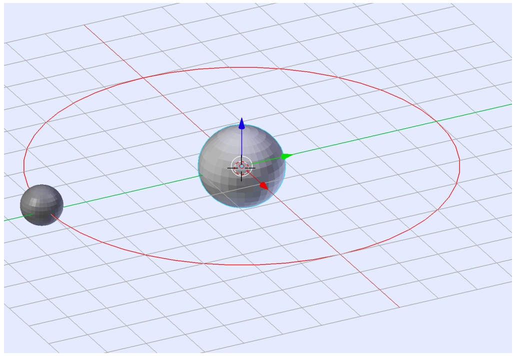


Massachusetts Institute of Technology

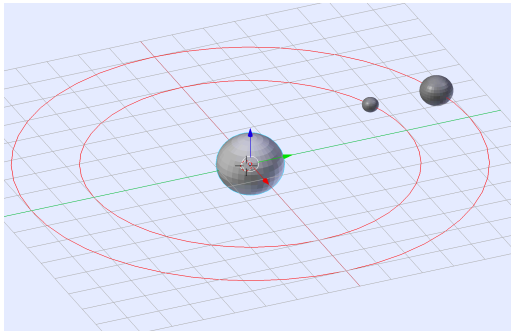
Generative Approach

Generate Physical Model

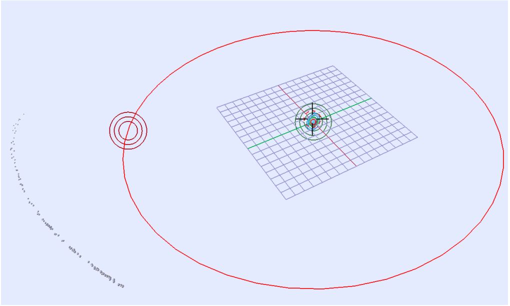
Scenario: One planet



Scenario: Two planets



Scenario: Irregular Orbiting Debris



Generative Approach: One Planet

Theoretical Domain

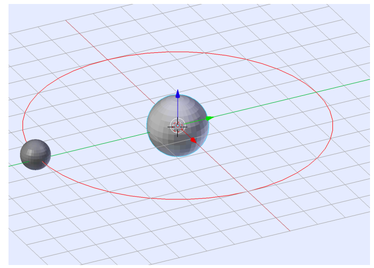
Physics

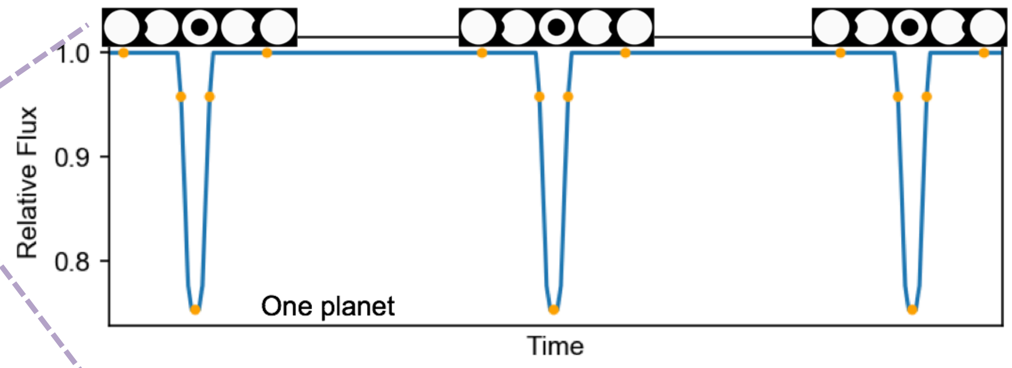
Model Features

Data Features

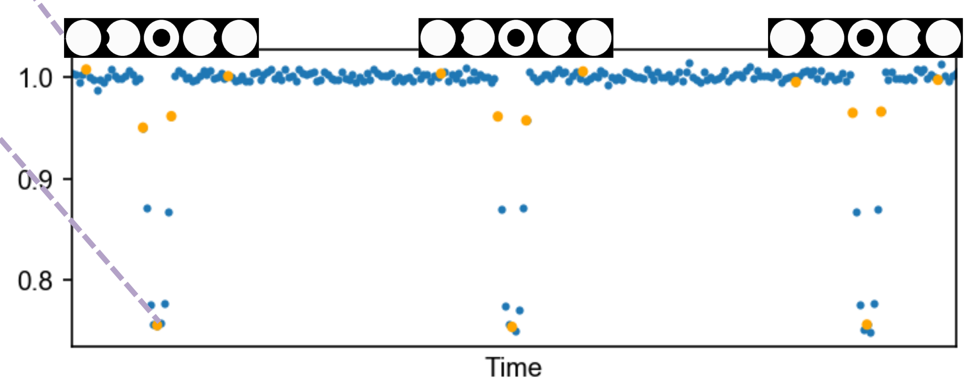
Empirical Domain

A “Rosetta Stone” linking models & theories to data

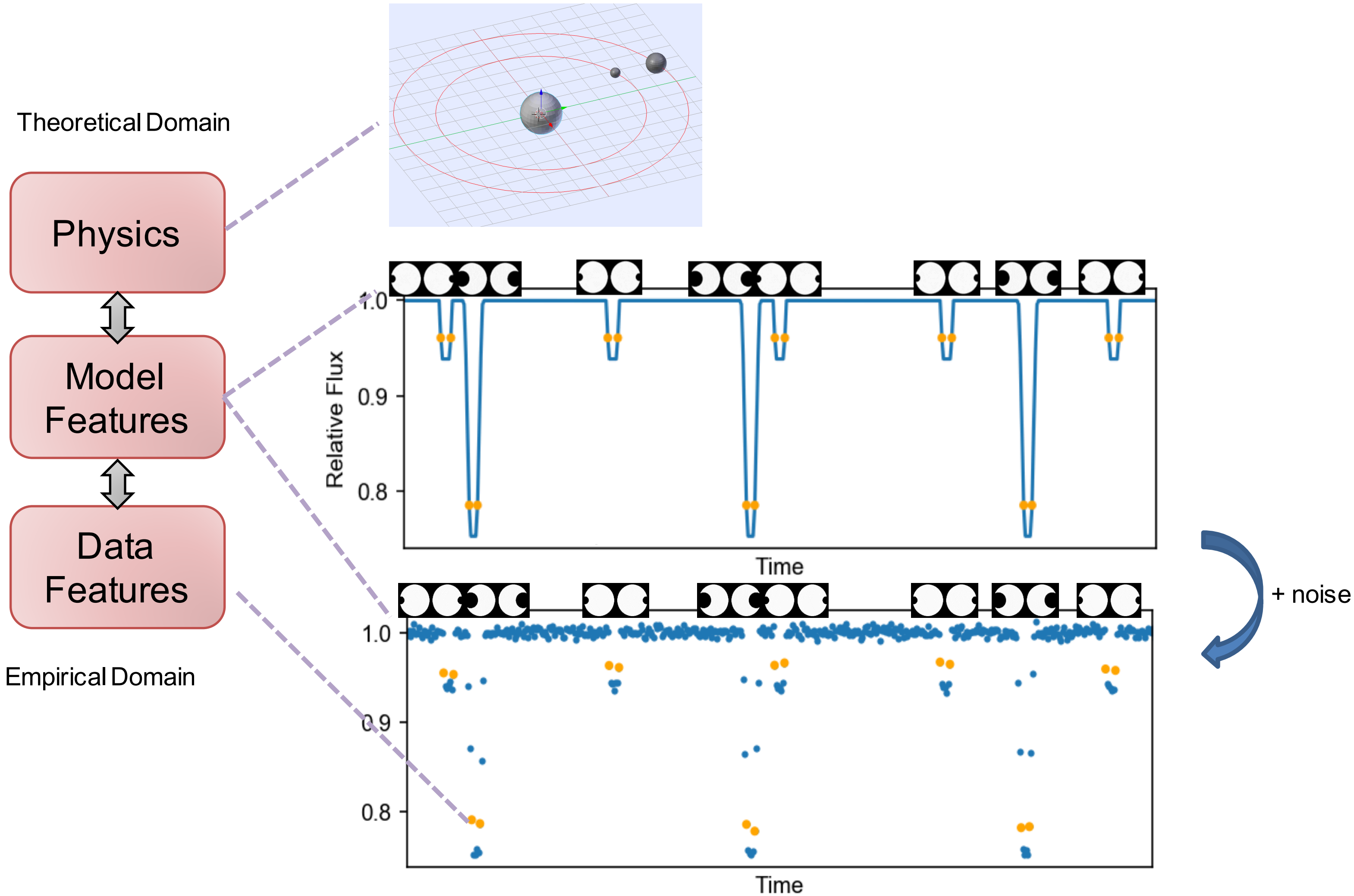




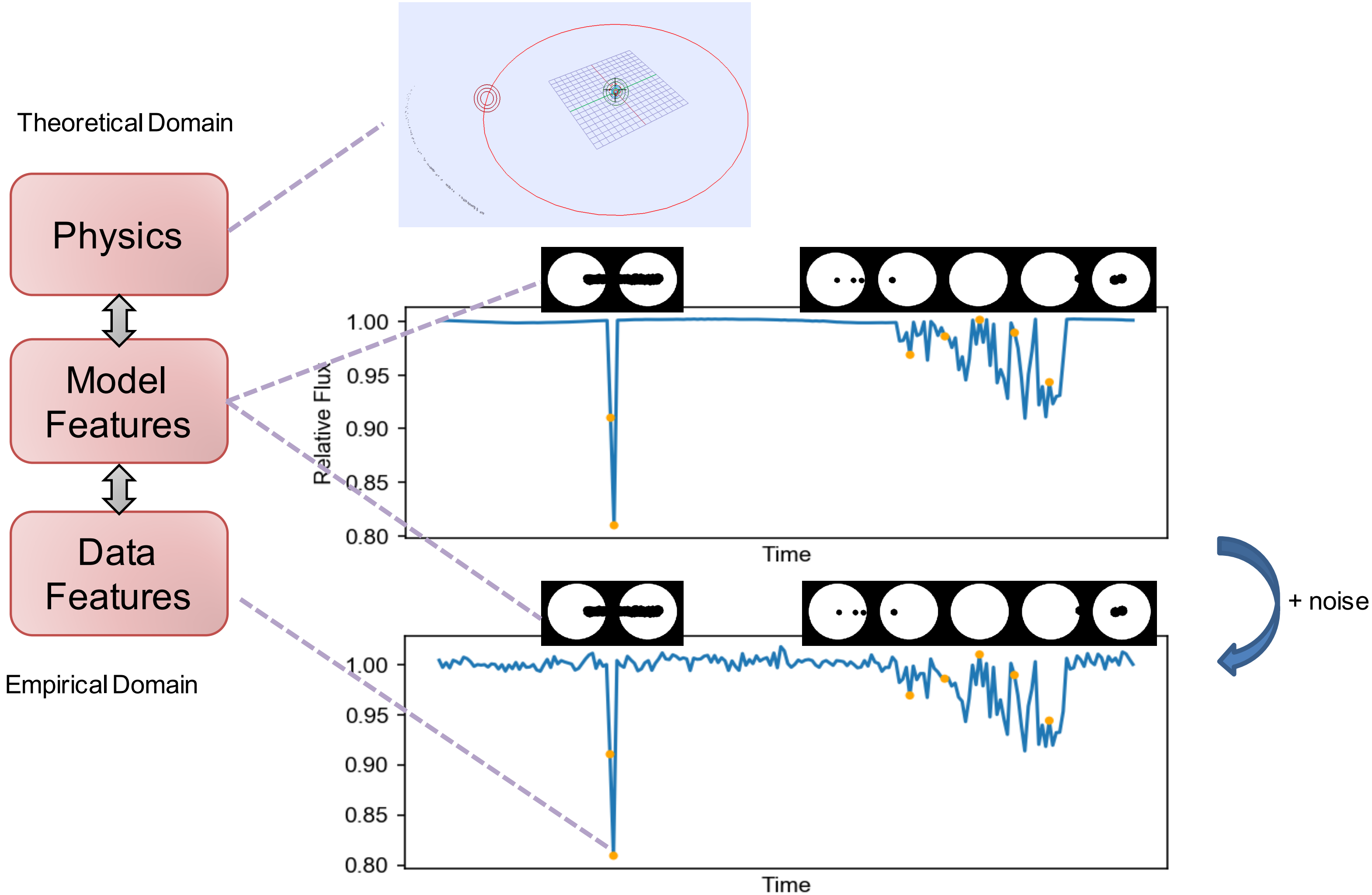
+ noise



Generative Approach: Two Planets

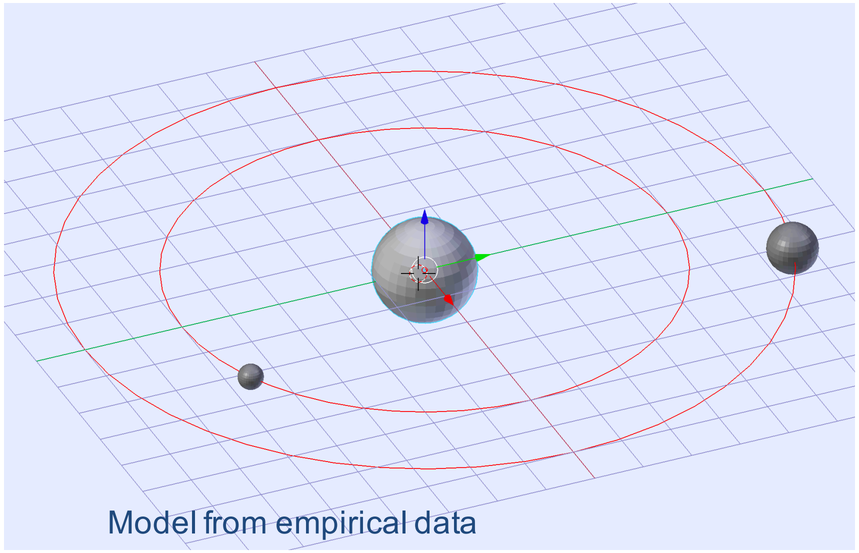


Generative Approach: Irregular Debris



Adding Inference Capabilities

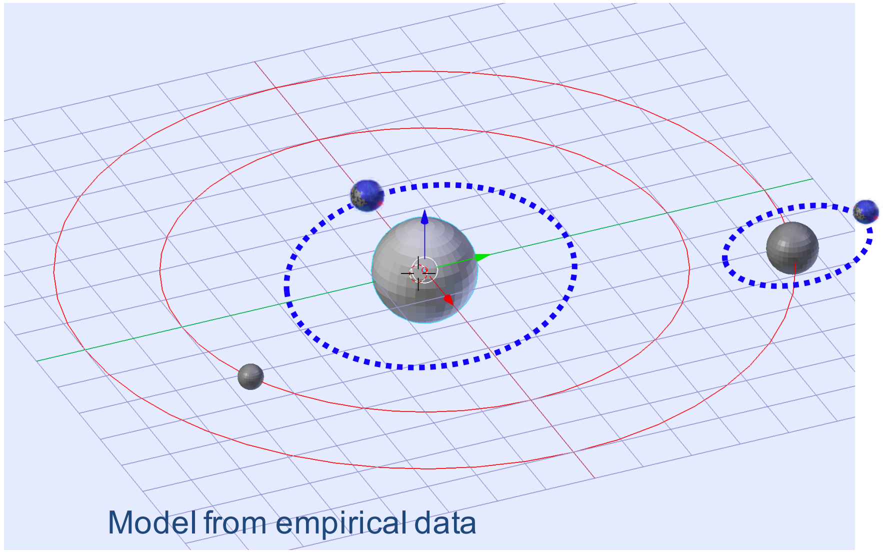
A system with a confirmed planet might have other planets, moons, debris disks, …



→ create an “autocomplete” capability (inference engine) for planetary systems

Adding Inference Capabilities

A system with a confirmed planet might have other planets, moons, debris disks, …



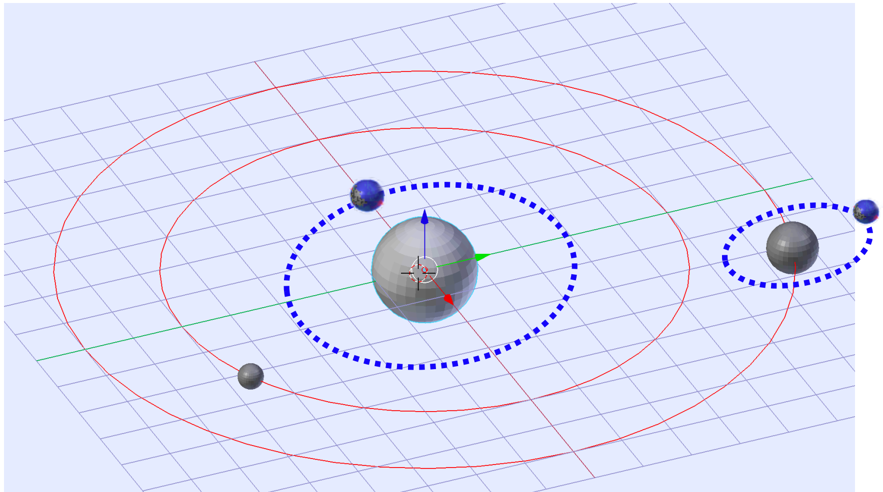
Model from empirical data

→ create an “autocomplete” capability (inference engine) for planetary systems

→ “Guess where & what” with plausible physics

Adding Inference Capabilities

A system with a confirmed planet might have other planets, moons, debris disks, …



Model from empirical data

→ create an “autocomplete” capability (inference engine) for planetary systems

→ “Guess where & what” with plausible physics

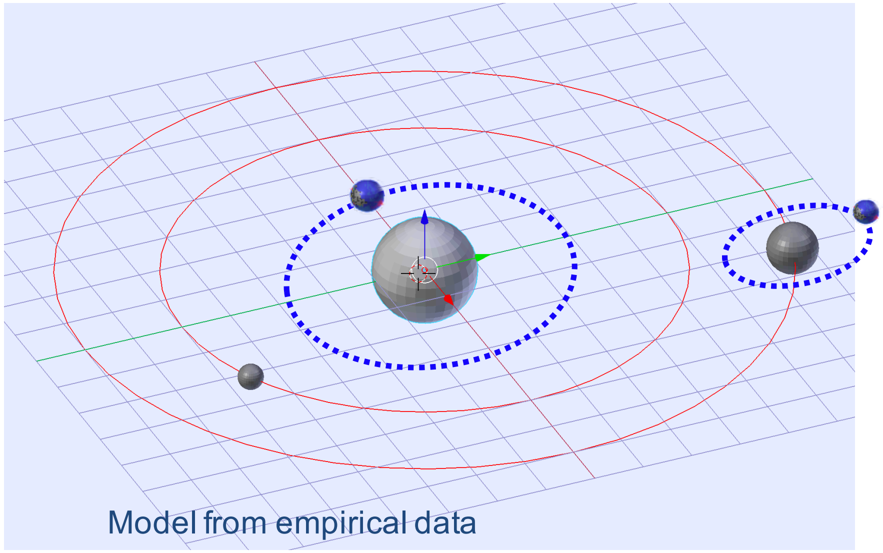
→ Create a population of forward models and plausible variants (e.g., using genetic programming)

Derive empirical features to look for, if models were describing reality

- Generate neural networks that have higher attention in those areas
- Test / falsify multiple theories in parallel

Massachusetts Institute of Technology

Adding Inference Capabilities



Generative approach facilitates inference on other properties

Planet mass, radius, orbital parameters, rotation rate, obliquity
⇒ gravitational acceleration
⇒ atmosphere parameters
⇒ potential mean density/rockiness
⇒ inferences on core, magnetosphere.

Planet surface temperature
⇒ greenhouse warming
⇒ thermal emission
⇒ atmospheric gases and compositions.

Spectroscopy parameters
⇒ biosignatures, gases
⇒ indicator factors of habitability

Host star properties
⇒ luminosity/temperature, spectral type, activity, rotation rate, and flare activity
⇒ habitability

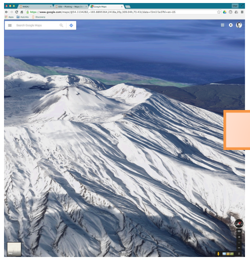
Massachusetts Institute of Technology

Can this approach can be transferred to other domains?

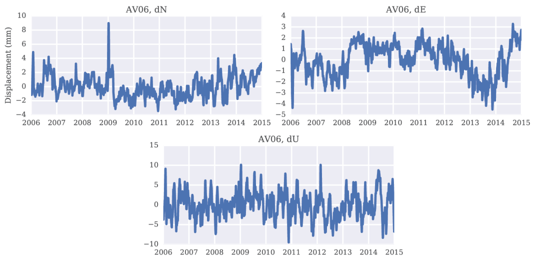
Geophysics Example

Volcanology

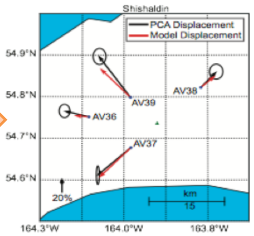
GPS Sensors



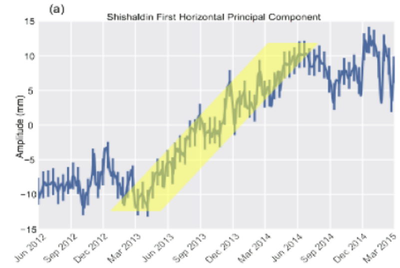
Time Series



Empirical Model

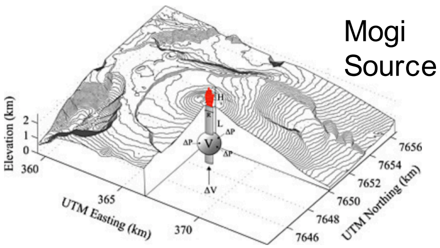


Classifier for
Earth deformation/ inflation event



Theoretical Model

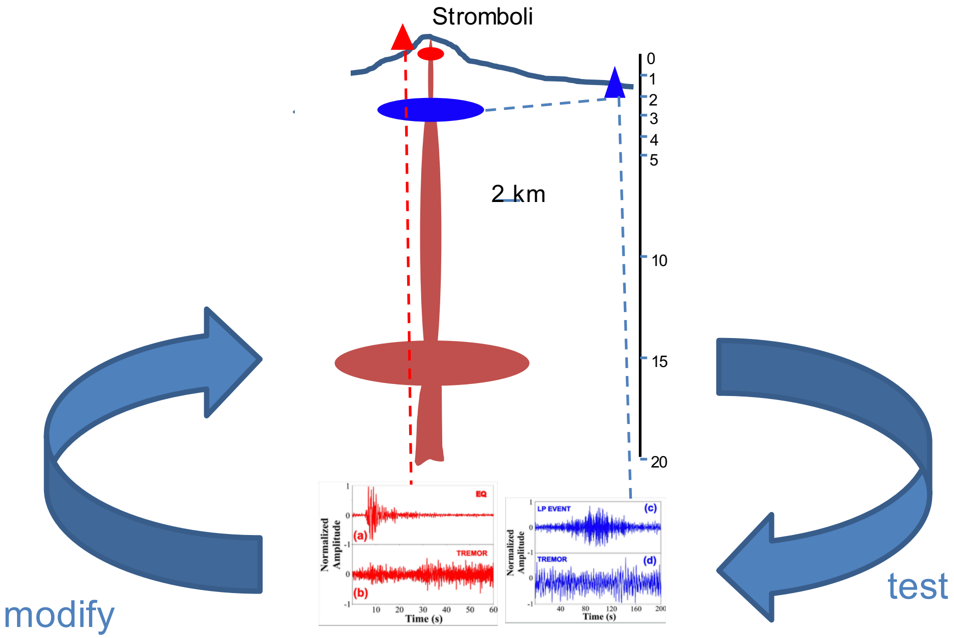
Mogi
Source



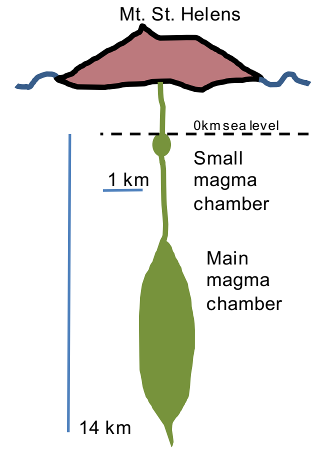
[J.Li, C.Rude, D.Blair, M.Gowanlock, T.Herring, V.Pankratius. Journal of Volcanology and Geothermal Research, 2016]

[Hibert et al., GRL ‘15]

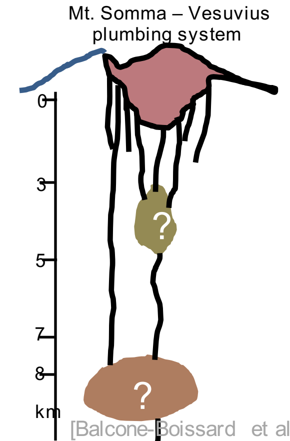
Inferring Models at Higher Abstraction Levels



[Giovanetti et al. Remote Sens., 8(4), 2016, Fig 3]



[Earle, Physical Geology, 2015, Fig. 4.12]

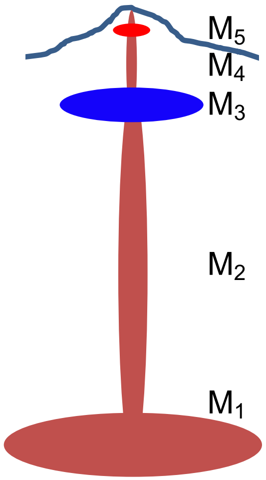


[Balcone-Boissard et al., Nature Scientific Reports 6, 21726, 2016; Fig. 1]

AI Theorem Prover for Science Models / Test Case Generator for Empirically Observable Features

- Derive test cases: “this property should be observable if this model was right”
- Derive falsification cases: “property that should never be observed if this model was right”
- Derive invariants: “this predicate should always be true if this model was right”

Symbolic Model Manipulation: Algebraic Approach



$$M_{seed} = M_1 \oplus M_2 \oplus M_3 \oplus M_4 \oplus M_5$$

perturb

$$\begin{aligned}\mathfrak{P}(M_{seed}) &= \mathfrak{P}(M_1 \oplus M_2 \oplus M_3 \oplus M_4 \oplus M_5) \\ &= \mathfrak{P}(M_1) \oplus \mathfrak{P}(M_2) \oplus \mathfrak{P}(M_3) \oplus \mathfrak{P}(M_4) \oplus \mathfrak{P}(M_5)\end{aligned}$$

M_i includes info on
- variables
- dom(variables)
- constraints(variables)

$$M_{1_1} \dots M_{1_n}$$

trim

$$\begin{aligned}\mathfrak{T}(M_{seed}) &= \mathfrak{T}(M_1 \oplus M_2 \oplus M_3 \oplus M_4 \oplus M_5) \\ &= M_1 \oplus M_2 \oplus M_3 \oplus M_4\end{aligned}$$

extend

$$\begin{aligned}\mathfrak{E}(M_{seed}, M_6) &= \mathfrak{E}(M_1 \oplus M_2 \oplus M_3 \oplus M_4 \oplus M_5; M_6) \\ &= M_1 \oplus M_2 \oplus M_3 \oplus M_4 \oplus M_5 \oplus M_6\end{aligned}$$

generate

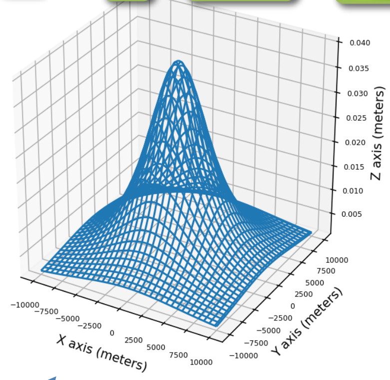
$$\mathfrak{G}(space(M)) = M_i \text{ with } M_i \in space(M)$$

Remark: more elaborate modeling requires introduction of a type system, constraints / domain-specific rules, …

[Pankratius et al., AGU'18]

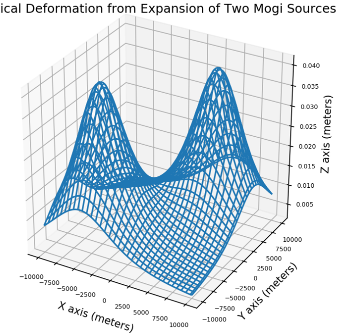
Examples for M_i in Geoscience

Mogi
Source Pos X
Source Pos Y
Source Depth
Amplitude
0
0
5km
1E6



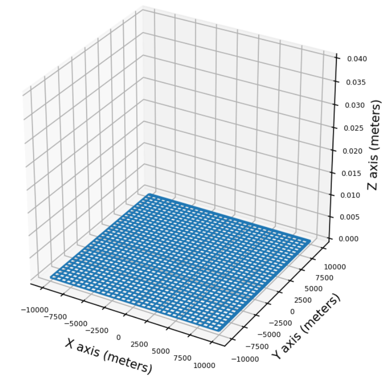
Composite
Mogi 1
Mogi 2
. . .
. . .

Vertical Deformation from Expansion of Two Mogi Sources

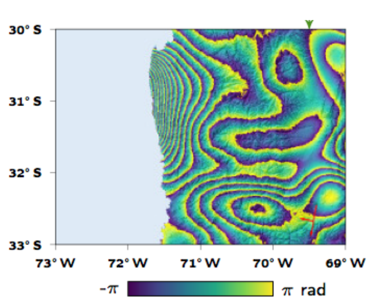


$\emptyset$

No deformation



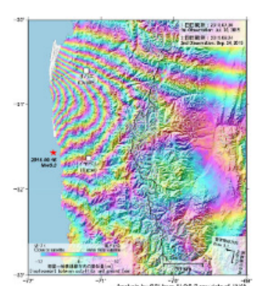
Compute Interferogram



add machine-learned noise components

Test with Reality

Compare with real-world InSAR satellite or UAV interferogram



Genetic Programming in Python, with a scikit-learn inspired API:

gplearn

```
est_gp = SymbolicRegressor(population_size=2000,
                           generations=20, stopping_criteria=1e-6,
                           model_set = model_set_minimal, const_dict=constants_dict,
                           p_crossover=0.1, p_subtree_mutation=0.1,
                           p_hoist_mutation=0.05, p_point_mutation=0.5,
                           max_samples=0.3, verbose=1,
                           parsimony_coefficient=0.0, random_state=2,
                           function_set=function_set, metric='rmse')
```

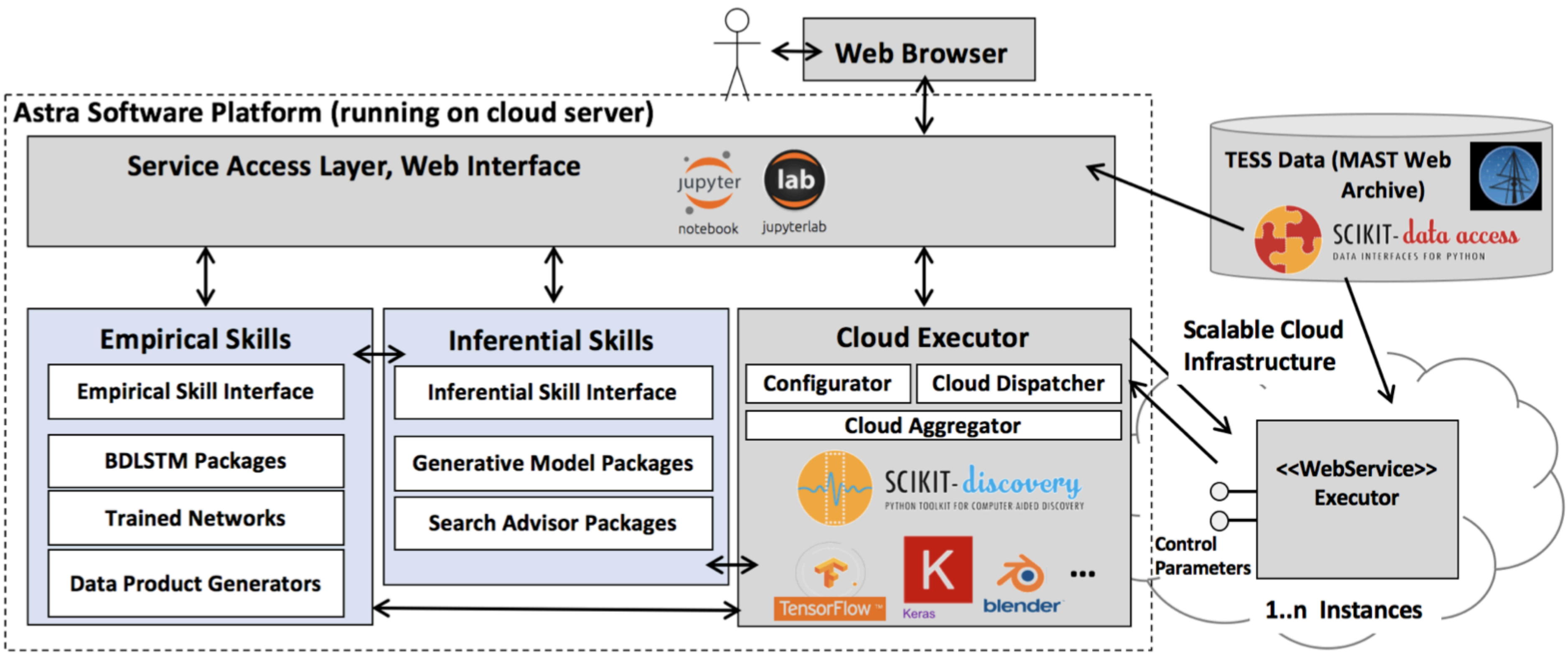
```
est_gp.fit(inputs,raveled_results);
```

```
    |    Population Average    |             Best Individual              |
---- ------------------------- ------------------------------------------ ----------
 Gen   Length          Fitness   Length          Fitness      OOB Fitness  Time Left
   0    29.29   0.379876288789        7 0.00520790406466 0.00505486366858      5.84m
   1      7.8   0.030539641708        7 0.00414515971233  0.0041535007502      3.94m
   2     7.47  0.0217428270721        7 0.00395724793119 0.00393784066313      3.10m
   3     7.54  0.0211037854184        7 0.00318999426958 0.00317289165736      2.65m
   4     7.46  0.0220503883175        7 0.00230268888262 0.00238928755349      2.32m
   5      7.4  0.0254420907836        7 0.00221451369295 0.00221619043055      2.07m
   6     7.56  0.0264524039272        7 0.00137761712883 0.00141688526414      1.85m
   7     7.58  0.0268504367133        7 0.00115849684897 0.00117389812701      1.67m
   8     7.38  0.0223381746746        7 0.00115217180138 0.00117656259719      1.50m
   9     7.56  0.0251189315923        7 0.00108114916971 0.00108388332304      1.34m
  10     7.34  0.0164327114159        7 0.00106194685183 0.00109198779104      1.18m
  11     7.48  0.0219102240383        7 0.000698760418825 0.000708090443757      1.04m
  12     7.43  0.0224319079123        7 0.000695351954025 0.000709526792845     53.96s
  13     7.56  0.0260644565465        7 0.000680539227613 0.000715654687798     45.86s
  14     7.42  0.0224007926631        7 0.000672186500833 0.000688974221301     37.89s
  15      7.4  0.0189147300504        7 0.000659537013899 0.000655411966447     30.09s
  16     7.44   0.020894681919        7 0.000648583111273 0.00066007953655     22.42s
  17     7.44  0.0209206195977        7 0.00064154116245 0.000663021882061     14.85s
  18     7.29  0.0159319391861        7 0.000638331590463 0.000664348006707      7.38s
  19     7.52  0.0192420494408        7 0.000637164640365 0.000659453373537      0.00s
```

[Rude, Pankratius, Rongier: work in progress]

- Where do we go from here?

Blueprint for "Astra"
An AI Science Assistant with Domain Knowledge



Massachusetts Institute of Technology

Conclusion

- Big Data & instrument fusion in scientific applications
  → push for more automation at all levels
- We need to rethink automation in the scientific process
- Problems go beyond detection, classifications, statistics
- Automated insight generation will be key
- Vision for future:
  AI science assistants that have domain knowledge

Thanks!

@vpankratius

pankrat@mit.edu

victorpankratius.com

AIST16 80NSSC17K0125
PI Pankratius

ACI1442997
PI Pankratius

Acknowledgement: This material is based upon work supported by the NSF and NASA. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the view of the NSF or NASA.